REFERENCES

1. A. Endres, 1975, "An analysis of errors and their causes in system programs", Proceedings of the International Conference on Reliable Software, pp. 327-336.
2. Cagatay Catal a, Ugur Sevima, Banu Diri b, 2011, "Practical development of an Eclipse-based software fault prediction tool using Naive Bayes algorithm", Elsevier Vol. 38, 2347–2353
3. Cederqvist P., 1992, Version management with CVS for CVS 1.11.2 p1, available at: http://cern-accelerators-optics.web.cern.ch/cern-accelerators/optics/cvs apache/cvs_manual.pdf.
4. D. Richard Kuhn, Dolores R. Wallace, 2004, "Software Fault Interactions and Implications for Software Testing" IEEE transactions on software engineering, vol. 30, no. 6.
5. Daniel Galin, 2004, Software Quality Assurance, for theory to implementation. Pearson Education Limited.
6. D.E. Rumelhart, G.E. Hinton, and R.J. Williams, 1986, "Learning internal representations by error propagation", in D.E. Rumelhart and J.L. McClelland [Eds.], Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1, MIT Press, Cambridge MA, pp. 318 - 362.
7. Deeprasertkul, P., Bhattarasinee, P., 2003, Software Fault Detection in C Programs, 12th Int. Conf. on Intelligent and Adaptive Systems and Software Engineering.
8. F. Akiyama, 1972, "An example of software system debugging", Inform. Processing, Proceedings of IFIP Congr., pp. 353-359.
9. Ghezzi, C., Jazayeri, M., Mandrioli, D., 2003. Fundamentals of Software Engineering. Prentice-Hall (International edition).
10. J.E. Gafhey, Jr., 1984, "Estimating the number of faults in code", IEEE Zhznsactions on Software Engineering, Vol. SE10, pp. 459-464.
11. Jay Naphas, 2011 "PREDICTING SOFTWARE FAULTS", Federal Aviation Administration/AST-300, 800 Independence Avenue SW, Washington, DC 20591, USA.
12. K. H. An, D. A. Gustafson, and A. C. Melton, 1987, A model for software maintenance," in Proceedings of the Conference on Software Maintenance, Austin, Texas. pp. 57-62, IEEE Computer Society Press.
13. L. Hatton, 1997, reexamining the fault density component size connection," IEEE Software, pp. 89-97.
14. Malhotra J and Tiple B, 2008, Software Testing and Quality Assurance. Nirali Prakashan, 2nd edition.
15. M. M. Baig and A. A. Khan, 2011, "A formal software testing techniques" Pakistan Journal of Science (Vol. 63 No. 4, pp 194 - 196).
16. M. Lipow, 1982, "Number of faults per line of code", IEEE Transactions on Software Engineering, Vol. SE8, pp. 437-439.
17. Mei-Huei Tang, Ming-Hung Kao, Mei-Hwa Chen, 1999, "An Empirical Study on Object-Oriented Metrics" SUNY at Albany Albany, NY 12222 (meitang, kao, mhc)@cs.albany.edu
18. M.H. Halstead, 1977, Elements of Software Science. New York: Elsevier.
19. M.L. Shooman, and M.I. Bolsky, 1975, "Types, distribution, and test and correction times for programming errors", Proceedings of 1975 Int. Conf. Rel. Software, pp. 347-362.
20. Maurizio Pighin, Vili Podgorelec, Peter Kokol, 2003, Fault threshold prediction with linier programming methodologies, Empirical software engineering 8, 117-138@kluwer academic publisher. Manufactured in the Netherlands.

21. McCabe, T. (1976). A complexity measure. IEEE Transactions on Software Engineering, 2(4), 308–320
22. R.H. Myers, 1990, Classical and Modern Regression with Applications. Duxbury Press, Boston, MA, USA,.
23. S. McConnell, 1997, "Gauging software readiness with defect tracking", IEEE Software, Vol. 14, no. 3, pp. 136, 135,.
24. S.C. Narula, J.F. Wellington, 1977, "Prediction of linear regression and the minimum sum of relative errors", Technometrics, Vol. 19, pp. 185- 190.
25. S. G. Eick, C. R. Loader, M. D. Long, L. G. Votta, and S. Vander Wiel, 1992, Estimating software fault content before coding," in Proceedings of the 14th International Conference on Software Engineering, Melbourne, Australia, pp. 59{65.
26. S.R. Chidamber, and C.F, Kemerer, 1994, "A Metrics Suite for Object Oriented Design", IEEE Transactions on Software Engineering, vol. 20, pp. 476-493,.
27. Supreet Kaur, and Dinesh Kumar, 2012, "Software Fault Prediction in Object Oriented Software Systems Using Density Based Clustering Approach" International Journal of Research in Engineering and Technology (IJRET) Vol. 1 No., ISSN: 2277-4378.
28. T. M. Khoshgoftaar, A.S. Pandya, and H.B. More, 1992, "A neural network approach for predicting software development faults", Proceedings of Third International Symposium on Software Reliability Engineering, pp. 83-89,.
29. T.M. Khoshgoftaar, J.C. Munson, 2006 "Predicting software development errors using software complexity metrics", IEEE Journal on Selected Areas in Communications, Vol. 8, issue 2, pp. 253-261.
30. T. M. Khoshgoftaar, a.s. Pandya, and h.b. More, 1992, "Prediction of Software Readiness Using Neural Network", ICITA pp. 114-123.
31. Taghi m. Khoshgoftaar, naeem seliya, 2010, Evolutionary Optimization of Software Quality Modeling with Multiple Repositories , (vol. 36 no. 6) pp. 852-864
32. T. J. Yu, V. Y. Shen, and H. E. Dunsmore, 1988 "An analysis of several software defect models", IEEE Transaction on Software Engineering, vol. 14, no. 9, pp. 1261-1270.
33. T. M. Khoshgoftaar, E. B. Allen, J. P. Hudepohl, S. J. Aud, 1997, "Application of neural networks to software quality modeling of a very large telecommunications system", Neural Networks, IEEE Transactions neural networks ,Vol. 8, Issue 4, pp. 902 – 909.
34. V. R. Basili and B. T. Perricone, 1984, "Software errors and complexity: An empirical investigation", Communications of the ACM, vol. 27, no. 1, pp. 42-52.
35. V. Shen, T. Yu, S. Thebaut and L. Paulsen, 1985, "Identifying error-prone software - an empirical study", IEEE Transactions of Software Engg., Vol. 11, no. 4, 317-324.
36. Tong-Seng Quah, Mie Mie Thet Thwin, 2004, "Prediction of Software Development Faults in PL/SQL Files Using Neural Network Models", Information and Software Technology, vol. 46, No.8, pp. 519-523.
37. Tong-Seng Quah And Mie Mie Thet Thwin, 2007, "Utilizing Computational Intelligence to Assist in Software Release Decision", Engineering Letters, Vol. 14 Issue 2, pp 89-94.
38. Yogesh Singh, Arvinder Kaur, Ruchika Malhotra, 2009, "Software Fault Proneness Prediction Using Support Vector Machines", Proceedings of the World Congress on Engineering Vol. 1, London, U.K.